UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/092,328	03/06/2002	David A. Carlson	005655.P007	9037

8791 7590 01/18/2006

BLAKELY SOKOLOFF TAYLOR & ZAFMAN
12400 WILSHIRE BOULEVARD
SEVENTH FLOOR
LOS ANGELES, CA 90025-1030

EXAMINER	
CERVETTI, DAVID GARCIA	
ART UNIT	**PAPER NUMBER**
2136	

DATE MAILED: 01/18/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/092,328	CARLSON, DAVID A.	
	Examiner	Art Unit	
	David G. Cervetti	2136	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *04 November 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-44* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-44* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Applicant's arguments filed November 4, 2005, have been fully considered but they are not persuasive.

2. Claims 1-44 are pending and have been examined.

Response to Amendment

3. The reply filed on November 4, 2005 ignores the objection to the specification that appeared on the prior Office Action.

4. Schneier clearly teaches "wherein a load operation associated with the generating of at least one portion of the ciphertext executes prior to a store operation associated with the generating of a prior portion of the ciphertext" (page 248). Schneier teaches on page 248 to unroll a loop and to overlap the start of one iteration with the end of a previous one (pipelining). The concept of pipelining instruction execution is extremely conventional and well known in the art. Pipelines are used to reduce the time to execute instructions. Executing instructions typically happens as loading the instruction into memory, reading registers, calculate an address or execute the operation, access an operand in memory, and write the output. Furthermore, speculative execution was also a conventional concept at the time the invention was made, and it would have been obvious to use processors that exploited such concept to perform encryption, much like Ye et al. do on the CHIMERA architecture.

5. Assuming arguendo Schneier does not expressly disclose the claimed language, Schneier not only provides the architecture where to implement the claimed language, but also provides the motivation to perform faster encryption and better utilization of

processing cycles to make the claimed language obvious to someone of ordinary skill in the art. Furthermore, Applicant appears to admit on page 14 of the "Remarks" that the **only alleged** difference between the instant application and the prior art of record is begin doing something **prior to** finish doing something else. Such argument is at the core of the concept of pipelining, which Schneier teaches on pages 242-259.

6. Applicant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the references.

7. Applicant's arguments do not comply with 37 CFR 1.111(c) because they do not clearly point out the patentable novelty which he or she thinks the claims present in view of the state of the art disclosed by the references cited or the objections made. Further, they do not show how the amendments avoid such references or objections.

Specification

8. The disclosure is objected to because of the following informalities: "execution of at least certain of the memory accesses" (page 10, paragraph 39, line 5). Appropriate correction is required.

Claim Rejections - 35 USC § 102

9. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

10. Claims 1, 4-5, 7, 13, 17-19, 21, 23, 28, 30, 32, 35-36, 38 are rejected under 35 U.S.C. 102(b) as being anticipated by Schneier et al. (NPL Fast Software Encryption: Designing Encryption Algorithms for Optimal Software Speed on the Intel Pentium Processor).

Regarding claim 1, Schneier et al. teach a computer-implemented method comprising: receiving a data cipher operation (pages 246-248); and processing the data cipher operation, wherein the processing comprises generating a number of portions of ciphertext from plaintext, wherein a load operation associated with the generating of at least one portion of the ciphertext executes prior to a store operation associated with the generating of a prior portion of the ciphertext (page 248, second paragraph).

Regarding claim 4, Schneier et al. teach wherein the store operation comprises swapping data within a data structure, the data within the data structure used in generating the ciphertext (pages 254-258).

Regarding claim 5, Schneier et al. teach wherein the load operation comprises accessing data from the data structure (pages 254-258).

Regarding claim 7, Schneier et al. teach wherein the data cipher operation comprises an RC4 operation (pages 246-248) and wherein the data structure comprises a substitution-box (pages 246-248).

Regarding claim 13, Schneier et al. teach a memory to store a data structure (pages 246-248); and a processing unit coupled to the memory, the processing unit to execute a data ciphering operation, wherein the processing unit is to swap data stored in the data structure for data ciphering of a first portion of plaintext, and wherein, prior to

the completion of the swapping of the data stored in the data structure for data ciphering of the first portion of the plaintext, the processing unit is to access data stored in the data structure for data ciphering of a second portion of the plaintext (page 248, second paragraph).

Regarding claim 17, Schneier et al. teach wherein the memory is to store the plaintext (pages 246-248).

Regarding claim 18, Schneier et al. teach wherein the data ciphering operation comprises an RC4 operation (pages 246-248).

Regarding claim 19, Schneier et al. teach wherein the data structure comprises a substitution-box (pages 246-248).

Regarding claim 21, Schneier et al. teach an interface unit to retrieve a data encryption operation, a substitution (S)-box and plaintext associated with the data encryption operation from the host memory based on an instruction from the host processor (pages 246-248); and an execution unit coupled to the interface unit, the execution unit comprising, a memory to store the plaintext and the S-box associated with the operation for the data cipher; a microcontroller unit to schedule the data cipher operation; and a RC4 unit to receive the data cipher operation, wherein the RC4 unit is to swap data stored in the S-box for data ciphering of a first portion of the plaintext and wherein the RC4 unit is to read data stored in the S-box for data ciphering of a second portion of the plaintext, prior to completion of the swapping of data stored in the S-box for data ciphering of the first portion of the plaintext (pages 246-248).

Regarding claim 23, Schneier et al. teach wherein the RC4 unit is to data cipher the first portion of the plaintext (page 248).

Regarding claim 28, Schneier et al. teach a host processor; a host memory coupled to the host processor, the host memory to include a security operation, wherein the security operation includes a data cipher operation based on RC4, the host memory to include plaintext and a data structure for the data cipher operation (pages 246-248); a co-processor coupled to the host processor, the co-processor comprising, an interface unit to retrieve the security operation from the host memory based on an instruction from the host processor; an execution unit coupled to the interface unit, the execution unit comprising, a memory to store the plaintext and the data structure associated with the data cipher operation; a microcontroller unit to store the data cipher operation in an execution queue; and an RC4 unit coupled to the execution queue, the RC4 unit to receive the data cipher operation, wherein the RC4 unit is to swap data stored in the S-box for data ciphering of a first portion of the plaintext and wherein the RC4 unit is to read data stored in the S-box for data ciphering of a second portion of the plaintext, prior to completion of the swapping of data stored in the S-box for data ciphering of the first portion of the plaintext (pages 246-248).

Regarding claim 30, Schneier et al. teach wherein the RC4 unit is to data cipher the first portion of the plaintext (page 248).

Regarding claim 32, Schneier et al. teach receiving a data cipher operation (pages 246-248); and processing the data cipher operation, wherein the processing comprises generating a number of portions of ciphertext from plaintext, wherein a load

operation associated with the generating of at least one portion of the ciphertext executes prior to a store operation associated with the generating of a prior portion of the ciphertext (page 248, second paragraph).

Regarding claim 35, Schneier et al. teach wherein the store operation comprises swapping data within a data structure, the data within the data structure used in generating the ciphertext (pages 254-258).

Regarding claim 36, Schneier et al. teach wherein the load operation comprises accessing data from the data structure (pages 254-258).

Regarding claim 38, Schneier et al. teach wherein the wherein the data cipher operation comprises an RC4 operation (pages 246-248) and wherein the data structure comprises a substitution-box (pages 246-248).

Claim Rejections - 35 USC § 103

11. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

12. Claims 2-3, 15-16, 20, 33-34 are rejected under 35 U.S.C. 103(a) as being unpatentable over Schneier et al.

Regarding claim 2, Schneier et al. do not disclose expressly wherein the generating of the at least one portion of the ciphertext and the generating of the prior portion of the ciphertext is executed within one iteration of a number of iterations for the data cipher operation. Schneier et al. teach using pipelines and parallelism for

instruction execution (pages 243-246, 254-258). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to execute the generating of the at least one portion of the ciphertext and the generating of the prior portion of the ciphertext within one iteration of a number of iterations for the data cipher operation. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to execute multiple instructions within one cycle.

Regarding claim 3, Schneier et al. do not disclose expressly wherein the generating of the at least one portion of the ciphertext is re-executed in a iteration that is subsequent to the one iteration upon determining that data retrieved from the load operation conflicts with data stored in the store operation. Schneier et al. teach using pipelines and parallelism for instruction execution, which deal with avoiding collisions and stalls (pages 243-246, 254-258). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to repeat an instruction if it was determined that there was a conflict. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to repeat an instruction if it was determined that there was a collision.

Regarding claim 15, Schneier et al. do not disclose expressly wherein the processing unit is to execute the data ciphering operation across a number of iterations, wherein the swapping of data stored in the data structure for data ciphering of the first portion of plaintext and the accessing of data stored in the data structure for data ciphering of the second portion of the plaintext are executed within one iteration of the number of iterations. Schneier et al. teach using pipelines and parallelism for instruction

execution (pages 243-246, 254-258). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to execute the generating of the at least one portion of the ciphertext and the generating of the prior portion of the ciphertext within one iteration of a number of iterations for the data cipher operation. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to execute multiple instructions within one cycle.

Regarding claim 16, Schneier et al. do not disclose expressly wherein the processing unit is to re-execute, within a subsequent iteration of the number of iterations, the accessing of data stored in the data structure for data ciphering of the second portion of the plaintext, upon determining that the data swapped for data ciphering of the first portion of plaintext equals the data accessed for the data ciphering of the second portion of the plaintext. Schneier et al. teach using pipelines and parallelism for instruction execution, which deal with avoiding collisions and stalls (pages 243-246, 254-258). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to repeat an instruction if it was determined that there was a conflict. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to repeat an instruction if it was determined that there was a collision.

Regarding claim 20, Schneier et al. do not disclose expressly wherein the apparatus is coupled to a host processor and a host memory, wherein the processing unit is to receive the data ciphering operation from the host memory. However, Examiner takes Official Notice that a processor receiving a data ciphering operation

from the host memory was conventional and well known. Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to have a processing unit receive a data ciphering operation from a host memory since Examiner takes Official Notice that it was conventional and well known.

Regarding claim 33, Schneier et al. do not disclose expressly wherein the generating of the at least one portion of the ciphertext and the generating of the prior portion of the ciphertext is executed within one iteration of a number of iterations for the data cipher operation. Schneier et al. teach using pipelines and parallelism for instruction execution (pages 243-246, 254-258). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to execute the generating of the at least one portion of the ciphertext and the generating of the prior portion of the ciphertext within one iteration of a number of iterations for the data cipher operation. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to execute multiple instructions within one cycle.

Regarding claim 34, Schneier et al. do not disclose expressly wherein the generating of the at least one portion of the ciphertext is re-executed in a iteration that is subsequent to the one iteration upon determining that data retrieved from the load operation conflicts with data stored in the store operation. Schneier et al. teach using pipelines and parallelism for instruction execution, which deal with avoiding collisions and stalls (pages 243-246, 254-258). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to repeat an instruction if it was determined that there was a conflict. One of ordinary skill in the art

would have been motivated to do so because it was well known in the art to repeat an instruction if it was determined that there was a collision.

13. Claims 6, 8-12, 14, 22, 24, 29, 31, 37, 39-44 are rejected under 35 U.S.C. 103(a) as being unpatentable over Schneier et al., and further in view of Puziol et al. (US Patent Number: 5,454,117).

Regarding claim 6, Schneier et al. do not disclose expressly wherein the generating of the at least one portion of the ciphertext is aborted upon determining that the data being swapped equals the data being accessed in the data structure. However, Puziol et al. teach backing up the machine state upon mispredicted branches (columns 1-2). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to abort generation of a portion of ciphertext upon determining that the data being swapped equals the data being accessed in the data structure. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to back up the machine state upon determining that it was not the correct data to act upon.

Regarding claim 8, Schneier et al. teach a computer-implemented method executing in a processor, the method comprising: receiving a request to perform for data ciphering of plaintext (pages 246-248); and processing the request based on a data structure stored in a memory coupled to the processor (pages 246-248), wherein the processing comprises, performing a first access of data from the data structure; swapping the data from the first access; data ciphering a first portion of the plaintext based on the swapped data from the first access; performing a second access of data

from the data structure prior to the swapping of the data from the first access (pages 246-248). Schneier et al. do not disclose expressly performing the following, upon determining that the data from the first access does not equal the data from the second access, swapping the data from the second access; and data ciphering a second portion of the plaintext based on the swapped data from the second access. However, Puziol et al. teach backing up the machine state upon mispredicted branches (columns 1-2). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to verify the data prior to processing it. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to perform data validation prior to processing data.

Regarding claim 9, the combination of Schneier et al. and Puziol et al. teach the limitations as set forth under claim 8 above. Furthermore, Schneier et al. teach using pipelines and parallelism for instruction execution (pages 243-246, 254-258).

Regarding claim 10, the combination of Schneier et al. and Puziol et al. teach the limitations as set forth under claim 9 above. Furthermore, Puziol et al. teach backing up the machine state upon mispredicted branches (columns 1-2).

Regarding claim 11, the combination of Schneier et al. and Puziol et al. teach the limitations as set forth under claim 8 above. Furthermore, Schneier et al. teach wherein the data ciphering comprises an RC4 operation (pages 246-248).

Regarding claim 12, the combination of Schneier et al. and Puziol et al. teach the limitations as set forth under claim 8 above. Furthermore, Schneier et al. teach wherein the data structure comprises a substitution-box (pages 246-248).

Regarding claim 14, Schneier et al. do not disclose expressly wherein the processing unit is to data cipher the second portion of the plaintext upon determining that the data being swapped in the data structure does not equal the data being accessed in the data structure. However, Puziol et al. teach backing up the machine state upon mispredicted branches (columns 1-2). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to verify the data prior to processing it. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to perform data verification prior to processing data.

Regarding claim 22, Schneier et al. do not disclose expressly wherein the RC4 unit is to data cipher the second portion of the plaintext upon determining that the data being swapped in the S-box does not equal the data being read from the S-box. However, Puziol et al. teach backing up the machine state upon mispredicted branches (columns 1-2). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to verify the data prior to processing it. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to perform data validation prior to processing data.

Regarding claim 24, Schneier et al. do not disclose expressly wherein the RC4 unit is to swap data retrieved from the S-box for the data ciphering of the second portion of the plaintext upon determining that the data being swapped for the data ciphering of the first portion of the plaintext does not equal the data read from the S-box for data ciphering of the second portion of the plaintext. However, Puziol et al. teach backing up

the machine state upon mispredicted branches (columns 1-2). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to verify the data prior to processing it. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to perform data validation prior to processing data.

Regarding claim 29, Schneier et al. do not disclose expressly wherein the RC4 unit is to data cipher the second portion of the plaintext upon determining that the data being swapped in the data structure does not equal the data being read from the data structure. However, Puziol et al. teach backing up the machine state upon mispredicted branches (columns 1-2). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to verify the data prior to processing it. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to perform data validation prior to processing data.

Regarding claim 31, Schneier et al. do not disclose expressly wherein the RC4 unit is to swap data retrieved from the data structure for the data ciphering of the second portion of the plaintext upon determining that the data being swapped for the data ciphering of the first portion of the plaintext does not equal the data read from the data structure for data ciphering of the second portion of the plaintext. However, Puziol et al. teach backing up the machine state upon mispredicted branches (columns 1-2). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to verify the data prior to processing it. One of ordinary skill in

the art would have been motivated to do so because it was well known in the art to perform data validation prior to processing data.

Regarding claim 37, Schneier et al. do not disclose expressly wherein the generating of the at least one portion of the ciphertext is aborted upon determining that the data being swapped equals the data being accessed in the data structure. However, Puziol et al. teach backing up the machine state upon mispredicted branches (columns 1-2). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to abort generation of a portion of ciphertext upon determining that the data being swapped equals the data being accessed in the data structure. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to back up the machine state upon determining that it was not the correct data to act upon.

Regarding claim 39, Schneier et al. teach receiving a request to perform data ciphering of plaintext (pages 246-248); and processing the request based on a data structure stored in a memory coupled to the processor (pages 246-248), wherein the processing comprises, performing a first access of data from the data structure; swapping the data from the first access; data ciphering a first portion of the plaintext based on the swapped data from the first access; performing a second access of data from the data structure prior to the swapping of the data from the first access (pages 246-248). Schneier et al. do not disclose expressly performing the following, upon determining that the data from the first access does not equal the data from the second access, swapping the data from the second access; and data ciphering a second

portion of the plaintext based on the swapped data from the second access. However, Puziol et al. teach backing up the machine state upon mispredicted branches (columns 1-2). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to verify the data prior to processing it. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to perform data validation prior to processing data.

Regarding claim 40, the combination of Schneier et al. and Puziol et al. teach the limitations as set forth under claim 39 above. Furthermore, Schneier et al. teach using pipelines and parallelism for instruction execution (pages 243-246, 254-258).

Regarding claim 41, the combination of Schneier et al. and Puziol et al. teach the limitations as set forth under claim 40 above. Furthermore, Puziol et al. teach backing up the machine state upon mispredicted branches (columns 1-2). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to re-execute the accessing of the data, swapping, and data ciphering upon a mispredicted branch. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to backing up machine state upon mispredicted branches.

Regarding claim 42, the combination of Schneier et al. and Puziol et al. teach the limitations as set forth under claim 39 above. Furthermore, Schneier et al. teach wherein the data ciphering comprises an RC4 operation (pages 246-248).

Regarding claim 43, the combination of Schneier et al. and Puziol et al. teach the limitations as set forth under claim 39 above. Furthermore, Schneier et al. teach wherein the data structure comprises a substitution-box (pages 246-248).

Regarding claim 44, the combination of Schneier et al. and Puziol et al. teach the limitations as set forth under claim 39 above. Furthermore, Schneier et al. teach using pipelines and parallelism for instruction execution (pages 243-246, 254-258). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to encrypt the plaintext over a number of iterations and encrypt the first portion of the plaintext in the same iteration of the encryption of the second portion of the plaintext. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to execute multiple instructions within one cycle.

14. Claims 25-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Batcher (US Patent Number: 6,873,707), and further in view of Puziol et al.

Regarding claim 25, Batcher teaches an apparatus comprising: a memory to store a substitution (S)-box (column 8, lines 47-67); an RC4 hardware state machine coupled to the memory to generate a plurality of output text blocks from a plurality of input text blocks, wherein a subset of said plurality of output text blocks are generated as a result of repeating the same sequence of states, wherein during each of the repeated sequence of states data is read from said S-box in said memory as part of the generation of a next one of said plurality of output text blocks prior to a write to said S-box in said memory completing as part of generation of a current one of said plurality of output text blocks (columns 9-10). Batcher does not disclose expressly wherein the data

is speculative read from said S-box. Puziol et al. teach speculative execution (columns 6-8). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to use speculative execution. One of ordinary skill in the art would have been motivated to do so because it was well known in the art to use branch prediction and dynamic scheduling to increase throughput.

Regarding claim 26, the combination of Batcher and Puziol et al. teach the limitations as set forth under claim 25 above. Furthermore, Batcher teaches wherein said plurality of output text blocks are ciphertext blocks and said plurality of input text blocks are plaintext blocks (column 8, lines 47-67).

Regarding claim 27, the combination of Batcher and Puziol et al. teach the limitations as set forth under claim 25 above. Furthermore, Batcher teaches wherein said plurality of input text blocks are ciphertext blocks and said plurality of output text blocks are plaintext blocks (column 8, lines 47-67).

Conclusion

15. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

16. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. US Patent 5,835,599 to Buer teaches multi-cycle non-parallel encryption, US Patent 5,801,975 to Thayer et al. teach processors performing speculative execution of instructions, Ye et al.'s NPL "CHIMAERA: A High-Performance Architecture with a Tightly-Coupled Reconfigurable Functional Unit" teaches speculative execution related to encryption.

17. Any inquiry concerning this communication or earlier communications from the examiner should be directed to David G. Cervetti whose telephone number is (571) 272-5861. The examiner can normally be reached on Monday-Friday 7:00 am - 5:00 pm, off on Wednesday.

18. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ayaz R. Sheikh can be reached on (571) 272-3795. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

19. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Primary Examiner
AU 2131
1/13/06

DGC